

**Amendments to the Specification:**

Please replace the paragraph beginning at page 1, line 4 through page 2, line 6, with the following amended paragraph:

The present application claims priority to U.S. Provisional Patent Application Serial No. 60/247,842, entitled Insertion Point Bungee Space Tool, filed November 10, 2000, which is hereby incorporated by reference as to its entirety. ~~The present application is also related to U.S. Provisional Application Serial No. 60/247,182, entitled Method and Apparatus For Improving the Appearance of Digitally Represented Handwriting, filed on November 10, 2000; to U.S. Provisional Application Serial No. 60/247,841, entitled Highlevel Active Pen Matrix, and filed on November 10, 2000; to U.S. Provisional Application Serial No. 60/247,973, entitled Selection Handles in Editing Electronic Documents, and filed on November 10, 2000; to application Serial No. (Atty docket 3797.00083), also entitled Selection Handles in Editing Documents, and filed on January 24, 2001; to U.S. Provisional Application Serial No. 60/247,844, entitled Simulating Gestures of a Mouse Using a Stylus and Providing Feedback Thereto, and filed on November 10, 2000; to U.S. Provisional Application Serial No. 60/247,400, entitled System and Method For Accepting Disparate Types Of User Input, and filed on November 10, 2000; to U.S. Provisional Application Serial No. 60/247,972, entitled In Air Gestures, and filed on November 10, 2000; to U.S. Provisional Application Serial No. 60/247,831, entitled Mouse Input Panel Windows Class List, and filed November 10, 2000; to U.S. Provisional Application Serial No. 60/247,843, entitled Mouse Input Panel and User Interface, and filed November 10, 2000; to U.S. Provisional Application Serial No. 60/247,479, entitled System and Method For Inserting Implicit Page Breaks, and filed on November 10, 2000; to application Serial No. (Atty. docket 3797.00086), entitled High Level Active Pen Matrix, and filed on December 15, 2000; to application Serial No. 09/741,107, entitled Mode Hinting/Switching, and filed on December 21, 2000; to U.S. Provisional Application Serial No. 60/247,847, entitled Tablet Computer and its Features, and filed on November 10, 2000; and to Application Serial No. 09/750,288, entitled Anchoring, Rendering, Reflow, & Transformations, filed December 29, 2000, each of which is incorporated by reference herein as to their entireties.~~

Please replace the paragraph beginning at page 3, line 9 through page 3, line 22, with the following amended paragraph:

However, the use of such a cursor for various editing actions requires the user to study and learn multiple functions and commands, thus complicating the user's interaction with the system. To illustrate, Figure ~~3a~~3A shows a screen of text from a prior art word processing program. In the Figure ~~3a~~3A screen, several lines of text 301 may have been generated by the user, for example, by typing the letters on a keyboard. If the user should desire to insert additional space, or text, at a point 302 within the lines of text 301, the user generally is required to position the cursor 303 at the insertion point, and then add spaces or text (by hitting the space bar or keys on a keyboard) to perform the insertion. Carriage returns may also be inserted by pressing the "Enter" or "Return" key on a conventional keyboard. However, with each press of a key (e.g., spacebar), the entire portion of the lines of text 301 beyond the cursor 303 shifts as a result of the addition. If the user wishes to insert 20 spaces, the text might shift 20 times for the spaces. This shifting is disorienting to the user, and complicates the user's interaction with the displayed document.

Please replace the paragraph beginning at page 4, line 1 through page 4, line 19, with the following amended paragraph:

A similar problem is encountered if the user wishes to "close up" space between two groups of text. To illustrate, Figure ~~3b~~3B shows another screen of text from a typical word processing program, in which two groups of text 304a, 304b are separated by space 305. If the user wishes to "close up" this space 305 by bringing the two groups of text together, the user typically must either: 1) highlight and delete the spaces and/or carriage returns between the groups of text; or 2) position the cursor at some point within the space 305, and begin pressing "Delete" or "Backspace" keys to delete the various components of space 305 (which may include spaces, tabs, carriage returns, etc.). These operations, however, also complicate the user's computer experience. In option 1, the user must learn how to properly select the entire space area 305, and then learn how to delete or cut the selected space area 305 from the document. An inadvertent selection of more, or less, than space 305 might result in an accidental deletion of

some or all of the groups of text 304a, 304b, requiring the user to backtrack and attempt the deletion again. Option 2 involves the deletion of individual spaces, tabs, carriage returns, etc., one at a time. This one-at-a-time deletion may be disorienting to the user as text group 304b may be shifted once for each deletion. On slower machines, the repeated deletions may be affected faster than they are displayed, resulting in over\_deletions. On faster machines, the rate at which pages scroll is faster than the user's ability to respond, further frustrating the user.

Please replace the paragraph beginning at page 4, line 20 through page 5, line 12, with the following amended paragraph:

Figures ~~3e-f~~3C-F illustrate another prior art attempt at easing the user's interaction with the displayed document. In the Microsoft® Inkwriter® product, a user could generate multiple lines 330 of text, which could be handwritten using a stylus, and ~~interacts~~interact with the displayed document using a pointing device (such as the stylus, mouse, etc.). The user could also place a cursor 332 at an insertion point. The cursor 332 would initially have the appearance of an "I-beam," as shown in Fig. ~~3e~~3C. As the user moved a pointer, such as a mouse or stylus pointer 334, over the cursor 332, also known as "hovering" over the cursor 332, a new hover symbol 336 would appear below the cursor 332, as shown in Figure ~~3d~~3D. If the user were to subsequently "click and drag" the mouse cursor by, for example, pressing a mouse button and moving the mouse, the hover symbol 336 would be replaced by an expanding arrow 338. This "click and drag" may also be performed using a stylus. As the user "dragged" the cursor 332, the expanding arrow 338 would then dynamically indicate the vertical position of the mouse pointer 334, as shown in Figure ~~3e~~3E, while maintaining the same horizontal position. Upon completing the drag, the text following the cursor 332 is moved to begin as a new line at the final vertical position of the mouse pointer 334.

Please replace the paragraph beginning at page 5, line 13 through page 6, line 2, with the following amended paragraph:

The Inkwriter® hover symbol 336 and expanding arrow 338 provide some feedback to the user, but this feedback is limited. For example, no additional feedback is provided if the user drags the cursor 332 within the same line (e.g., to the left or right). Additionally, the amount of

vertical feedback is limited to a vertical indication only. In the Figure ~~3e~~3E screen, the expanding arrow 338 does not indicate the horizontal positioning of the pointer 334, resulting in some confusion on the part of the user as to what will happen when the dragging ceases. The operation of the Inkwriter® cursor is also confusing when the user wishes to delete space in a document. In deleting space or text, the Inkwriter® cursor 332 behaves as described above with respect to Figures ~~3e-e~~3C-E, resulting in a feedback display as shown in Figure ~~3g~~3G. In the Figure ~~3g~~3G screen, the expanding arrow 338 similarly provides limited feedback regarding exactly what text, if any, will be deleted when the dragging is completed.

Please replace the paragraph at page 7, line 7 with the following amended paragraph:

Figures ~~3a-g~~3A-G depict prior art screens of text using prior art word processors.

Please replace the paragraph beginning at page 7, line 8 through page 7, line 9, with the following amended paragraph:

Figures ~~4a-4~~4A and ~~4b-4~~4B illustrate graphical user interface icons according to at least one embodiment of the present invention.

Please replace the paragraph beginning at page 7, line 10 through page 7, line 11, with the following amended paragraph:

Figures ~~5a-e~~5A-C illustrate directional flows for various written forms of communication, or languages.

Please replace the paragraph at page 7, line 12 with the following amended paragraph:

Figures ~~6a-e~~6A-E are screen shots using one or more aspects of the present invention.

Please replace the paragraph at page 7, line 13 with the following amended paragraph:

Figures ~~7a-e~~7A-C are screen shots using further aspects of the present invention.

Please replace the paragraph beginning at page 7, line 16 through page 7, line 2, with the following amended paragraph:

The present invention may be more readily described with reference to FIGS. 1-2 and 4-~~7e~~4-7. Figure 1 illustrates a schematic diagram of a conventional general-purpose digital computing environment that can be used to implement various aspects of the present invention. In Figure 1, a computer 100 includes a processing unit 110, a system memory 120, and a system bus 130 that couples various system components including the system memory to the processing unit 110. The system bus 130 may be any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of bus architectures. The system memory 120 includes read only memory (ROM) 140 and random access memory (RAM) 150.

Please replace the paragraph beginning at page 17, line 10 through page 17, line 21, with the following amended paragraph:

Since the insertion point 603 may be placed within text and then moved, a determination may be made as to whether text upstream or downstream from the point is to be moved. For example, the space created in the example shown in Figs. 6A-D is generated by moving the text that was originally downstream from point 603, further downstream. In alternative embodiments, the same space could have been generated by moving the text upstream from point 603 further upstream. In one embodiment, this determination may be made based on the direction of the movement of the moving cursor 604b. If the moving cursor 604b was moved downstream, then the text downstream from point 603 is moved further downstream. If the moving cursor 604b was moved upstream, the text upstream from point 603 would be moved further upstream. It will be understood that this determination may be made in other ~~was~~ways to generate the same space between the cursors 604a and 604b.